



*Full credit is given to the above companies including the OS that this PDF file was generated!*

## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'tracepath6.8' command***

***\$ man tracepath6.8***

TRACEPATH(8)                    iputils                    TRACEPATH(8)

### NAME

tracepath - traces path to a network host discovering MTU along this path

### SYNOPSIS

tracepath [-4] [-6] [-n] [-b] [-l pktlen] [-m max\_hops] [-p port] [-V]  
          {destination}

### DESCRIPTION

It traces the network path to destination discovering MTU along this path. It uses UDP port port or some random port. It is similar to traceroute. However, it does not require superuser privileges and has no fancy options.

tracepath -6 is a good replacement for traceroute6 and classic example of application of Linux error queues. The situation with IPv4 is worse, because commercial IP routers do not return enough information in ICMP error messages. Probably, it will change, when they are updated. For now it uses Van Jacobson's trick, sweeping a range of UDP ports to maintain trace history.

## OPTIONS

-4

Use IPv4 only.

-6

Use IPv6 only.

-n

Print primarily IP addresses numerically.

-b

Print both: Host names and IP addresses.

-l

Sets the initial packet length to pktlen instead of 65535 for IPv4  
or 128000 for IPv6.

-m

Set maximum hops (or maximum TTLs) to max\_hops instead of 30.

-p

Sets the initial destination port to use.

-V

Print version and exit.

## OUTPUT

```
root@mops:~ # tracepath -6 3ffe:2400:0:109::2
```

```
1?: [LOCALHOST]          pmtu 1500
```

```
1: dust.inr.ac.ru        0.411ms
```

```
2: dust.inr.ac.ru      asymm 1 0.390ms pmtu 1480
```

```
2: 3ffe:2400:0:109::2    463.514ms reached
```

```
Resume: pmtu 1480 hops 2 back 2
```

The first column shows the TTL of the probe, followed by colon. Usually the value of TTL is obtained from the reply from the network, but sometimes it does not contain the necessary information and we have to guess it. In this case the number is followed by ?.

The second column shows the network hop which replied to the probe. It is either the address of the router or the word [LOCALHOST], if the probe was not sent to the network.

The rest of the line shows miscellaneous information about the path to the corresponding network hop. It contains the value of RTT, and additionally it can show Path MTU when it changes. If the path is asymmetric or the probe finishes before it reaches the prescribed hop, the difference between number of hops in forward and return direction is shown next to the keyword "async". This information is not reliable, e.g. the third line shows asymmetry of 1. This is because the first probe with TTL of 2 was rejected at the first hop due to Path MTU Discovery.

The last line summarizes information about all the paths to the destination. It shows detected Path MTU, amount of hops to the destination and our guess about the number of hops from the destination to us, which can be different when the path is asymmetric.

## SEE ALSO

traceroute(8), traceroute6(8), ping(8).

## AUTHOR

tracepath was written by Alexey Kuznetsov <kuznet@ms2.inr.ac.ru>.

## SECURITY

No security issues.

This lapidary deserves to be elaborated. tracepath is not a privileged program, unlike traceroute, ping and other beasts of their kind. tracepath may be executed by everyone who has enough access to the network to send UDP datagrams to the desired destination using the given port.

#### AVAILABILITY

tracepath is part of iputils package.

iputils 20210202

TRACEPATH(8)